Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (currently amended): A method of binary instrumentation comprising:

using a processor, coupled to a memory, to allocate ~~allocating~~ a contiguous memory region included in the memory;

filling the memory region with at least one copy of an interceptor function;

initializing a first data structure with at least a starting address, length of the allocated memory region, and a reference to a second data structure;

storing an address of an original function in a current element of the second data structure upon request for instrumentation; and

providing a starting address of a copy of the interceptor function upon request for instrumentation.

Claim 2 (original): The method of claim 1, wherein allocating a memory region, filling the memory region, and initializing the first data structure are performed upon an initial request for instrumentation.

Claim 3 (original): The method of claim 1, wherein allocating a memory region, filling the memory region, and initializing the first data structure are performed if all interceptor function copies of currently allocated memory regions are associated with previous requests for instrumentation.

Claim 4 (original): The method of claim 1, further comprising duplicating the first data structure to associate each new copy of the first data structure with each newly allocated memory region.

Claim 5 (original): The method of claim 1, wherein the second data structure comprises elements to store addresses of original functions instrumentation was requested for.

Claim 6 (original): The method of claim 1, further comprising maintaining the current element of the second data structure to establish a correspondence between the original function and a provided address of an interceptor function copy.

Claim 7 (original): The method of claim 6, further comprising selecting a next successive element of the second data structure as the current element for each new request for instrumentation.

Claim 8 (original): The method of claim 1, wherein a reference to the second data structure comprises at least one of a memory address of and an index to the second data structure.

Claim 9 (original): The method of claim 1, wherein the starting address of a copy of the interceptor function is provided in a direct correspondence with the current element of the second data structure.

Claim 10 (original): The method of claim 1, wherein the interceptor function comprises

obtaining an address being currently executed;

retrieving, from a corresponding copy of the first data structure, the starting address of a memory region that contains the address being currently executed;

fetching the reference to the second data structure from the copy of the first data structure;

computing an index to the second data structure as the fetched reference to the second data structure added to the difference between the address being currently executed and the retrieved starting address, the difference divided by the size of the interceptor function; and

reading, from the second data structure indexed with the computed index, the address of an original function to pass control to.

Claim 11 (currently amended): An article comprising:

a machine accessible physical medium having a plurality of machine readable instructions, wherein when the instructions are executed by a processor, the instructions provide for binary instrumentation by:

allocating a contiguous memory region, wherein a total number of original functions to be intercepted is unknown at the time of the allocation;

filling the memory region with ~~at least one copy~~ (A) first and second platform independent high level non-assembly language copies of an interceptor function (B) but not any version of the original functions;

initializing a first data structure with at least a starting address, length of the allocated memory region, and a reference to a second data structure;

storing an address of [[an]] one of the original ~~function~~ functions to be intercepted in a current element of the second data structure upon request for instrumentation; and

providing a starting address of the second [[a]] copy of the interceptor function upon request for instrumentation.

Claim 12 (currently amended): The article of claim 11, wherein instructions for allocating [[a]] the memory region, filling the memory region, and initializing the first data structure are executed upon initial request for instrumentation so that each time an additional original function needs to be intercepted a new copy of the interceptor function is associated with the additional original function.

Claim 13 (currently amended): The article of claim 11, wherein instructions for allocating [[a]] the memory region, filling the memory region, and initializing the first data structure are executed only if all interceptor function copies of currently allocated memory regions are associated with previous requests for instrumentation.

Claim 14 (original): The article of claim 11, further comprising instructions for duplicating the first data structure to associate each new copy of the first data structure with each newly allocated memory region.

Claim 15 (original): The article of claim 11, wherein the second data structure comprises elements to store addresses of original functions instrumentation was requested for.

Claim 16 (original): The article of claim 11, further comprising instructions for maintaining the current element of the second data structure to establish a correspondence between the original function and a provided address of an interceptor function copy.

Claim 17 (original): The article of claim 16, further comprising instructions for selecting a next successive element of the second data structure as the current element for each new request for instrumentation.

Claim 18 (original): The article of claim 11, wherein a reference to the second data structure comprises at least one of a memory address of and an index to the second data structure.

Claim 19 (currently amended): The article of claim 11, wherein the starting address of the second [[a]] copy of the interceptor function is provided in a direct correspondence with the current element of the second data structure.

Claim 20 (original): The article of claim 11, wherein interceptor function comprises instructions for

obtaining an address being currently executed;

retrieving, from a corresponding copy of the first data structure, the starting address of a memory region that contains the address being currently executed;

fetching the reference to the second data structure from said copy of the first data structure;

computing an index to the second data structure as the fetched reference to the second data structure added to the difference between the address being currently executed and the retrieved starting address, said difference divided by the size of the interceptor function; and

reading, from the second data structure indexed with the computed index, the address of an original function to pass control to.

Claim 21 (currently amended): A system that performs binary instrumentation, comprising:

a plurality of unique copies of an interceptor function including a separate copy for each original function to be intercepted; and

a processor, coupled to a memory, to operate an instrumenting module to allocate a contiguous memory region included in the memory, to fill said memory region with the copies of the interceptor function, to initialize a first data structure with at least a starting address, length of the allocated memory region, and a reference to a second data structure, to store an address of an original function in a current element of the second data structure upon request for instrumentation, and to provide a starting address of a copy of the interceptor function upon request for instrumentation.

Claim 22 (currently amended): The system of claim 21, wherein (A) the instrumenting module is executed upon an initial request for instrumentation, (B) each one of the copies is not created until a corresponding request for instrumentation occurs such that the system is configured to dynamically adapt to a number of functions to be instrumented and (C) a total number of original functions to be intercepted is unknown at the time of the allocation.

Claim 23 (original): The system of claim 21, wherein the instrumenting module is executed if all interceptor function copies of currently allocated memory regions are associated with previous requests for instrumentation.

Claim 24 (original): The system of claim 21, wherein the instrumenting module duplicates the first data structure to associate each new copy of the first data structure with each newly allocated memory region.

Claim 25 (original): The system of claim 21, wherein the second data structure comprises elements to store addresses of original functions instrumentation was requested for.

Claim 26 (original): The system of claim 21, wherein the instrumenting module maintains the current element of the second data structure to establish a correspondence between the original function and a provided address of an interceptor function copy.

Claim 27 (currently amended): The system of claim 26, wherein the instrumenting module selects a next successive element of the second data structure as the current element for each new request for instrumentation and the copies are stored adjacent to one another.

Claim 28 (original): The system of claim 21, wherein reference to the second data structure comprises at least one of a memory address of and an index to the second data structure.

Claim 29 (original): The system of claim 21, wherein the starting address of a copy of the interceptor function is provided in a direct correspondence with the current element of the second data structure.

Claim 30 (original): The system of claim 21, wherein the interceptor function is adapted to

obtain an address being currently executed;

retrieve from a corresponding copy of the first data structure the starting address of a memory region that contains the address being currently executed;

fetch the reference to the second data structure from said copy of the first data structure;

compute an index to the second data structure as the fetched reference to the second data structure added to the difference between the address being currently executed and the retrieved starting address, said difference divided by the size of the interceptor function; and

read, from the second data structure indexed with the computed index, the address of an original function to pass control to.